APPLICATION UNDER UNITED STATES PATENT LAWS

Atty. Dkt. No. PW 0279249
<div align="center">(M#)</div>

Invention: EARLY WARNING IN E-SERVICE MANAGEMENT SYSTEMS

Inventor (s): Shi-Yue QIU

Pillsbury Winthrop LLP
Intellectual Property Group
1600 Tysons Boulevard

McLean, VA 22102
Attorneys
Telephone: (703) 905-2000

This is a:

☐ Provisional Application

☒ Regular Utility Application

☐ Continuing Application
　☐ The contents of the parent are incorporated
　　by reference

☐ PCT National Phase Application

☐ Design Application

☐ Reissue Application

☐ Plant Application

☐ Substitute Specification
Sub. Spec Filed _____
　　　in App. No. _____/_____

☐ Marked up Specification re
Sub. Spec. filed _____
　　　In App. No _____/_____

SPECIFICATION

EARLY WARNING IN E-SERVICE MANAGEMENT SYSTEMS

1. Application Data

[0001]   The present invention is related to five provisional patent applications: U.S. Patent Application No. 60/243,472, titled "The eService Business Model", U.S. Application No. 60/243,401, titled "Framework for eService Management", U.S. Patent Application No. 60/243,469, titled "Behavior Experts in eService Management", U.S. Application No. 60/243,397, titled "The Uniform Data Model", and U.S. Application No. 60/243,470, titled "Adaptive Feedback Control in eService Management". The present invention as well as the five provisional patent applications relate to various aspects of eService management. The subject matter of each is hereby incorporated by reference into each of the others.

2. Reservation of Copyright

BACKGROUND

3. Field of the Invention

[0003]   Aspects of the present invention relate to the field of e-commerce. Other aspects of the present invention relate to a method and system to intelligently manage an infrastructure that supports an e-service business.

4. General Background and Related Art

[0004]   The expanding use of the World-Wide Web (WWW) for business continues to accelerate and virtual corporations are becoming more commonplace. Many new businesses, born in this Internet Age, do not employ traditional concepts of physical site location (bricks and mortar), on-hand inventories and direct customer contact. Many traditional businesses, that want to survive the Internet revolution are rapidly reorganizing (or re-inventing) themselves into web-centric enterprises. In today's high-speed Business-to-Business (B2B) and Business-to-Customer (B2C) eBusiness environment, a business entity must provide high

quality service, scale to accommodate exploding demand and be flexible enough to rapidly respond to market changes.

[0005]   The growth of eBusiness is being driven by fundamental economic changes. Firms that harness the Internet as the backbone of their business are enjoying tremendous market share gains – mostly at the expense of the unenlightened that remain true to yesterday's business models. Whether it is rapid expansion into new markets, driving down cost structures, or beating competitors to market, there are fundamental advantages to eBusiness that cannot be replicated in the "brick and mortar" world.

[0006]   This fundamental economic shift, driven by the tremendous opportunity to capture new markets and expand existing market share, is not without great risks. If a customer cannot buy goods and services quickly, cleanly, and confidently from one supplier, a simple search will divulge a host of other companies providing the same goods and services. Competition is always a click away.

[0007]   eBusinesses are rapidly stretching their enterprises across the globe, connecting new products to new marketplaces and new ways of doing business. These emerging eMarketplaces fuse suppliers, partners and consumers as well as infrastructure and application outsourcers into a powerful but often intangible Virtual Enterprise. The infrastructure supporting the new breed of virtual corporations has become exponentially more complex – and, in ways unforeseen just a short while ago, unmanageable by even the most advanced of today's tools. The dynamic and shifting nature of complex business relationships and dependencies is not only particularly difficult to understand (and, hence manage) but even a partial outage among just a handful of dependencies can be catastrophic to an eBusiness' survival.

[0008]   Businesses are racing to deploy Internet enabled services in order to gain competitive advantage and realize the many benefits of eBusiness. For an eBusiness, time-to-value is so critical that often these business services are brought online without the ability to manage or sustain the service. eBusinesses have been ravaged with catastrophe after catastrophe. Adequate technology, to effectively prevent these catastrophes, does not exist.

[0009]   eBusiness infrastructures operate around the clock, around the globe, and constantly evolving. If a critical supplier in Asia cannot process an electronic order due to infrastructure problems, the entire supply chain comes to a grinding halt. Who understands

the relationships between technology and business processes and between producer and supplier? Are they available 24 hours / day, 7 days / week, and 365 days/year? How long will it take to find the right person and rectify the problem? The promise of B2B, B2C and eCommerce in general will not be fully realized until technology is viewed in light of business process to solve these problems.

[0010]   Web-enabled eBusiness processes effectively distill all computing resources down to a single customer-visible service (or eService). For example, a user interacts with a web site to make an online purchase. All of the back-end hardware and software components supporting this service are hidden, so the user's perception of the entire organization is based on this single point of interaction. How can organizations mitigate these risks and gain the benefits of well-managed eServices?

[0011]   Never before has an organization been so dependent on a single point of service delivery – the eService. An organization's reputation and brand depend on the quality of eService delivery because, to the outside world, the eService is the organization. If service delivery is unreliable, the organization is perceived as unreliable. If the eService is slow or unresponsive, the company is perceived as being slow or unresponsive. If the Service is down, the organization might as well be out of business.

[0012]   Further complicating matters, more and more corporations are outsourcing all or part of their web-based business portals. While reducing capital and personnel costs and increasing scalability and flexibility, this makes Application Service Providers (ASPs), Internet Service Providers (ISPs) and Managed Service Providers (MSPs) the custodians of a corporation's business. These "xSPs" face similar challenges – delivering quality service in a rapid, cost efficient manner with the added complication of doing so across a broad array of clients. Their ability to meet Service Level Agreements (SLAs) is crucial to the eBusiness developing a respected, high quality electronic brand – the equivalent of prime storefront property in a traditional brick and mortar business.

[0013]   The Internet enables companies to outsource those areas in which the company does not specialize. This collaboration strategy creates a loss of control over infrastructure and business processes between companies comprising the complete value chain. Partners, including suppliers and service providers must work in concert to provide a high quality service. But how does a company control infrastructure which it doesn't own

and processes that transcend its' organizational boundaries? Even infrastructure outsourcers don't have mature tools or the capability to manage across organizational boundaries.

[0014]   The underlying problem is not lack of resources, but the misguided attempt to apply yesterday's management technology to today's eService problem. As noted by Forrester Research, "Most companies use 'systems' management tools to solve pressing operational problems. None of these tools can directly map a system or service failure to business impact." To compensate, they rely on slow, manual deployment by expensive and hard-to-find technical personnel to diagnose the impact of infrastructure failures on service delivery (or, conversely, to explain service failures in terms of events in the underlying infrastructure). The result is very long time-to-value and an unresponsive support infrastructure. In an extremely competitive marketplace, the resulting service degradation and excessive costs can be fatal.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015]   The present invention is further described in terms of exemplary embodiments which will be described in detail with reference to the drawings. These embodiments are non-limiting exemplary embodiments, in which like reference numerals represent similar parts throughout the several views of the drawings, and wherein:

[0016]   Fig. 1 shows a high-level block diagram of an eService management system;

[0017]   Fig. 2 shows expanded block diagrams of both local service management systems and the global eService management system and their interactions via a dispatcher;

[0018]   Fig. 3 shows the input and output relationship of a Behavior eXpert (BeX);

[0019]   Fig. 4 shows different functional modes of a BeX;

[0020]   Fig. 5 illustrates an exemplary internal structure of a BeX in relation to other parts in a local service management system;

[0021]   Fig. 6 shows a time series variable values with an underlying pattern;

[0022]   Fig. 7 shows an exemplary variable behavior that can be described by two embedded patterns;

[0023]   Fig. 8 depicts the internal structure of the statistical learning mechanism of a BeX;

[0024]   Fig. 9 is an exemplary flowchart of a process, in which statistical models characterizing the normal and dynamic behavior of a variable are established and are applied in generating early warning of threshold violation in eService management;

[0025]   Fig. 10 is an exemplary flowchart for online normal behavior modeling;

[0026]   Fig. 11 illustrates the actual behavior of a time series variable and its violation of a threshold;

[0027]   Fig. 12 illustrates the predicted behavior of a time series variable; and

[0028]   Fig. 13 is an exemplary flowchart for an early warning mechanism.

DETAILED DESCRIPTION

[0029]   An embodiment of the present invention is illustrated that is related to Behavior eXperts (BeXs) employed in an eService management system. The present invention enables intelligent eService management by incorporating statistical behavior modeling and abnormal behavior forecasting (or early warning) capabilities in a BeX.

[0030]   A Behavior Expert (BeX) in an eService management system is a distributed, autonomous intelligent agent, designed to detect, analyze, predict, and control certain behavior of the components of a business infrastructure that supports the underlying eService. A BeX may be attached to a component (or an application) of an eBusiness infrastructure so that the operational status or the behavior of the component may be dynamically monitored and adaptively adjusted to optimize the eService quality.

[0031]   Fig. 1 is a high level diagram of an eService Management System 100. An eService 105 is a web-centric service, which allows electronic transactions over the Internet. Such a web-centric service may, for example, sell books, shoes, or flowers. It may also sell stocks or information. The eService 105 is supported by an eService infrastructure 115, which may comprise infrastructure components such as web servers, databases, billing systems, or other eServices. In the eService infrastructure 115, each component may play a distinct role. For example, for a shoes.com eService that sells shoes, a database may be part

of the infrastructure that supports shoes.com service and the database may store all the transaction information. The performance of each infrastructure component may affect the overall quality of service of shoes.com eService.

[0032]   In Fig. 1, there is a cluster, 110, of local service management systems. Each of the local service management systems may be responsible for the management of a local system which is part of the eService infrastructure 115. For example, local service management system 110b may be responsible for managing a database for an eService called shoes.com. A local system may comprise one or more infrastructure components. The performance information about infrastructure components or a local system of the eService infrastructure 115 may be sent, via a dispatcher 130, to a global data repository (not shown), located in a global eService management system 150. The information stored in the global data repository may be accessed and integrated by the global eService management system 150 to assess the overall performance of the eService infrastructure 115 and subsequently to estimate the overall service quality of the eService 105. In Fig. 1, the dispatcher 130 may represent a collective comprising one or more distributed dispatchers.

[0033]   The quality of an eService depends on various factors. Such factors are related to both the performance of individual infrastructure components and how the business process of the eService takes place within the supporting eService infrastructure. Different components in the eService infrastructure 115 may impact the quality of eService differently, depending on the role of each component with respect to the business process of the eService. Therefore, the strategy to manage the infrastructure that supports an eService may be directly related to or dictated by the business process model of the eService.

[0034]   In Fig. 1, business process model 120 is derived from the eService 105. The business process model 120 dictates both how the eService infrastructure 115 should be managed by local service management systems 110 and how the global eService management system 150 integrates the information from systems 110 to evaluate the overall performance of the eService infrastructure 115. The knowledge about the business process model 120 may be distributed in local service management systems 110a, 110b, ..., 110c.

[0035]   There may be multiple global eService management systems. Different global eService management systems may be responsible for different eServices but they may share local service management systems. Therefore, while the global eService management system

- 6 -

150 may seem to be a centralized unit in Fig. 1, it may be distributed, similar to local service management systems.

[0036]   Fig. 2 presents the exemplary internal structures of both a local service management systems (110b) and the global eService management system 150 and how they interact with each other. In Fig. 2, local service management system 110b comprises a plurality of data providers 210, a service manager 220, one or more Behavior eXperts (BeXs) 215, a local ecology pattern detector 225, an adaptive feedback control mechanism 230, and a communication unit 240.

[0037]   Data providers 210 supply observation data (observations in terms of, for example, the operational status), acquired from various infrastructure components, to the service manager 220. The service manager 220 converts the observation data to Generic Data Objects so that different Behavior eXperts (BeXs) 215 may access the observation data in a uniform way.

[0038]   Each BeX in a local service management system may be designated to monitor an infrastructure component. A BeX at component level may access the observation data acquired (by the data providers) from the underlying infrastructure component and analyze the behavior of the infrastructure component based on the observation data. A BeX may post some detected abnormal behavior of individual components, in the form, for example, states or events, on a blackboard server (not shown in Fig. 2) located in the service manager 220. Such posted information may be shared among different BeXs and accessed by the local ecology pattern detector 225.

[0039]   The local ecology pattern detector 225 may retrieve information from the blackboard server so that abnormal behavior occurred in different infrastructure components may be reviewed as a whole in order to detect any alarming trend or ecological pattern of the underlying local system. Detected ecological patterns may be reported, in the form of, for example, events together with some of the abnormal events at component level that have high priorities, to the dispatcher 130, via the communication unit 240.

[0040]   Each local service management system (110a,…,110b,…110c) may act asynchronously to monitor the performance of a local infrastructure. Internal to each local management system (110a,…,110b,…110c), an adaptive feedback control mechanism 230 may be activated so that the behavior of a local service management system may be

adaptively tuned towards some desired behavior. For example, if a BeX in a local service management system (110b) always reports a certain type of abnormal event and it always turned out to be a false alarm (e.g., the reported event does not have a significant ecological impact on the local system), the local service management system 110b may trigger the adaptive feedback control mechanism 230 to tune the responsible BeX so that the BeX becomes less sensitive to these events and, consequently, to become more aware of the events that actually do not impact the eService.

[0041]   The performance information gathered from different local service management systems may be routed, through the dispatcher 130, to the global eService management system 150. The global eService management system 150 comprises a global ecology controller 255, an eService enterprise 250, a design studio 260, a eService manager 270, a notifier 280, and a port 290 for external APIs.

[0042]   Data routed from the dispatcher 130 may be stored in the global data repository 245 and accessed by the global ecology controller 255. The global ecology controller 255 may then integrate the information from local service management systems 110 to and evaluate the performance of the overall eService infrastructure. The global ecology controller 255 may also estimate the service quality of the eService 105 based on the assessment about the overall infrastructure performance. This may be done by measuring the impact of detected abnormal behavior in different parts of the infrastructure on the eService. The translation from local infrastructure performance data to overall eService quality may be performed based on the business process model of the underlying eService.

[0043]   The global ecology controller 255 may also activate an adaptive feedback control. It may send feedback adjustments to different local service management systems, from where the adjustments may be passed further down to various individual BeXs. The purpose of activating an adaptive feedback control may be to tune the behavior of an eService management system so that it converges to an optimal state to ensure the quality of an eService.

[0044]   In Fig. 2, both the local ecology pattern detectors 225 as well as the global ecology controller 255 may be realized using BeXs. Essentially, a BeX is an intelligent reasoning mechanism that takes input data and generates inference output based on its expert knowledge. The distinction between a BeX at component level and a BeX for, for example,

realizing a local ecology pattern detector, may be merely functional rather than structural and methodological. A BeX that is attached to an infrastructure component may perform an individual monitoring task. A BeX implemented at an ecological level may perform higher level integration task.

[0045]   Fig. 3 depicts the input and output relationship of a BeX. A BeX 215 may be associated with one or more infrastructure components 310. Data providers 210 acquire performance data from the associated infrastructure components 310 and supply observation data to the BeX 215. To detect abnormal behavior in the associated components, the BeX may base its analysis on the observation data supplied by the data providers 210. When abnormal behavior is detected, the BeX throws one or more events 320 to signal the abnormal behavior of the underlying components 310. Events thrown by other BeXs may also be made available by the data providers 210 as the observation data. In this way, different BeXs may interact with each other, sharing what is detected and making further inferences.

[0046]   Fig. 4 illustrates that a BeX may function in different modes: learning mode 410 and operational mode 420. In Fig. 4, the observation data is fed to a BeX and may be utilized during both the learning mode 410 and the operational model 420. During the learning mode 410, the BeX learns the patterns of variables or ordinary behavior of the variables under normal operation environment of the system. Such learning may be achieved using different methods. In the exemplary construct of a BeX shown in Fig. 4, a statistical learning mechanism 430 is used to accomplish the task. The learned behavior may be captured in a behavior model of the variable. Such a model may be an linear or non-linear model.

[0047]   In the operational mode, a BeX monitors its associated component(s) and detects any abnormal behavior. Abnormal behavior may be defined a priori or it may be detected by comparing with learned normal behavior. Detection of abnormal behavior of an infrastructure component may be achieved by an operational mechanism 450 within a BeX. The operational mechanism 450 monitors the operational status of its associated component through the observation data and determines whether the operational status is acceptable according to some criteria. For example, a BeX that monitors a database may detect an abnormal behavior when the database is not responding to queries, given that the acceptable behavior of the database is that its responding time to a query should be less than 20 seconds.

In this case, the BeX reports the abnormal behavior after detecting that the normal responding time has elapsed.

[0048]   The variable behavior learned during the learning mode 410 may be applied during the operational mode to proactively predict any incoming abnormal behavior before it occurs. In Fig. 4, such proactive prediction is achieved by an early warning mechanism 440. Based on the learned variable behavior from the statistical learning mechanism 430 and the observation data (that reflect the current behavior of the underlying component), the early warning mechanism 440 estimates, with some certainty (may be expressed in the form of a probability), when, in the future, an abnormal behavior will occur. Such early warning may be sent to the operational mechanism 450 which will react accordingly to either report the estimated trend or incorporate the warning into its own inference.

[0049]   The learning mode 410 and the operational mode 420 may be running at different times or simultaneously. Particularly, during the learning mode 410, there may be different states of learning. For example, a BeX may learn some variable behavior offline from some historical data in a batch mode or the BeX may learn dynamic variable behavior online during its operation in an incremental fashion. The former may be applied before the BeX is first deployed and the latter may be applied after the BeX is up and running.

[0050]   In the operational mode, the designated tasks of a BeX are dictated through a set of variables and rules and the reaction of the BeX to the operational status of its underlying component is defined through a set of events. This is illustrated in Fig. 5. In Fig. 5, a BeX 215 operates based on variables 510, rules 520, and events 320. Rules 520 govern the transitional relationship between the variables 510 and events 320. Events 320 may be generated based on updated states which may be set based on the values of the variable 510. Rules 520 may be classified into metric rules and behavior rules, where the metric rules govern the transition between variables and states and behavior rules govern the transition between states and events.

[0051]   Observation data acquired by the data providers 210 is sent to a general data server 220a where the observation data is converted into Generic Data Objects (GDO) 220b so that heterogeneous kinds of data may be packaged and accessed in a uniformed way.

[0052]   A BeX (e.g., 215) may access the GDOs 220b to instantiate or to populate its internal variables 510. The updated variable values may trigger or fire rules 520. Fired rules

may then generate certain events 320 (indicating abnormal behavior of the infrastructure components that are monitored by BeX 215), which are formatted in accordance with the UDM 530 before being posted on the blackboard server 540.

[0053]   A rule may define some violation of acceptable behavior and may take the form:

name. IF premise THEN then-action ELSE else-action;

wherein "name" is the identifier of a particular rule, "IF premise" describes a condition, "then-action" describes the action to be taken when the condition satisfies, and "else-action" describes the action to be taken when the condition does not satisfy. The condition described in the "IF premise" may specify violation of acceptable behavior in terms of a variable value exceeds some expected value or threshold. For example, "IF Memory Capacity < 20%" describes that when the value of variable Memory Capacity is below a threshold of 20%" (a threshold that may define that the acceptable behavior of a memory is that it has more than 20% of its memory available), a violation of a threshold occurs. The rules may be designed to enforce some performance requirements, imposed on the running components of an eService infrastructure to support an underlying eService.

[0054]   Detecting abnormal behavior usually involve comparing variable values to some thresholds. Since the underlying infrastructure component that is monitored may operate continuously, the variable values may need to be sampled regularly according to some internal clock (which also regulates how often the BeX detects abnormal behavior in its operational mode). Such regular data sampling produces time series variables, each of which may present some particular pattern over time. The statistical learning mechanism 410 is designed to learn such patterns based on time series variable values.

[0055]   Various mathematical and statistical techniques are available for discovering sets of repeating patterns from collections of data. Using statistical learning, both short term and long term harmonic patterns in data and, knowing the regularity of the pattern (within error tolerances) can be used to predict the behavior at some future time. When a BeX is in its learning mode, it may continuously collect data from the associated data providers for several periods and performs statistical analysis to discover any emerging patterns in the data. Fig. 6 illustrates an example in which the time series values of a variable X form an emerging pattern over time. In Fig. 6, the horizontal axis represents time, vertical axis represents the magnitude of variable values, the dots represent the discrete values of a variable X recorded

over time, and the curve is a sine-wave like pattern representing the emerging patter of variable X in time.

[0056]   Data points recorded over time often include noise or outliers that are usually extraneous data points that do not fit into the principle pattern of the data. In detecting an emerging pattern based on recorded data points, such noise may have to be considered in the modeling process by either modeling the noise simultaneously or reducing the brittleness of the data prior to the modeling. By removing noise, the emerging patter or the actual trend line over the analysis time horizon may be more reliably discovered. This discovery may take the form of a non-linear model to capture the variable's behavior.

[0057]   Time series variables may have different underlying intrinsic patterns of varying amplitudes and wavelengths. A data stream containing only one or two patterns is called shallow data while data streams that have many patterns is called deep data. Fig. 7 illustrates a data stream that may be represented by two different underlying patterns embedded in the value of variable X. In Fig. 7, the first pattern, pattern 1, presents a high frequency and the second patter, pattern 2, presents a lower frequency. They are modulated on top of each other and together they form the underlying pattern of the variable X over time.

[0058]   A statistical learning model may be designed to identify and to quantify any number of such intrinsic patterns, although long term patterns with low amplitudes may be much more difficult to detect since they are generally obscured by random noises. Data series containing multiple patterns also introduce a higher level of noise (seen as apparent randomness or excessive outliers) into the modeling process simply by virtue of the patterns themselves. The modeling technique used to learn the behavior of variables that are characterized by multiple patterns may have to be designed accordingly to deal explicitly with problems associated with multiple and embedded patterns.

[0059]   The validity of a model and hence its predictive capabilities is fundamentally determined, all other factors being equal, by its access to historical data. For example, if inter-day behavior is needed, several days of data collection is necessary. However, if day-to-day variation pattern analysis within a week is also needed, several weeks of data collection is required. The amount of data required is proportional to both the longitudinal scope of the underling pattern and the necessary precision of the model itself.

- 12 -

[0060]   Fig. 8 depicts an exemplary construct of the statistical learning mechanism 430, which comprises two parts: an offline normal behavior modeling mechanism 810 and an online behavior modeling mechanism 820. The offline normal behavior modeling mechanism 810 learns a variable's normal pattern in a batch mode based on offline observation data corresponding to pre-recorded data points. What it captures is the static or regular pattern of the underlying variable without considering the dynamic noise factor. For example, a sine wave is a regular pattern that can be characterized using a sine function.

[0061]   The online behavior modeling mechanism 820 learns the dynamics of a variable's behavior based on online observation data corresponding to the data points collected during a BeX's operations. What it captures is the dynamic or adaptive pattern of the underlying variable, which is modulated on top of the regular pattern, learned during the offline modeling. For example, if a variable has, under normal situations, a sine pattern, its values measured online usually will not exactly fit the sine wave. This may be due to noise. To model a variable's pattern, both its regular and its dynamic patterns need to be captured. The online behavior modeling mechanism 820 is designed to characterize the variable's dynamics in time.

[0062]   Using what is learned by both the offline normal behavior modeling mechanism 810 and the online behavior modeling mechanism 820, a compound statistical model for a variable may be built that is capable of characterizing the real time behavior of a variable.

[0063]   The variable patterns that a BeX learns offline are the ordinary behaviors as seen under the (assumed) normal operation of the system. These behaviors may be encoded in a non-linear time series model. This model is deployed when the BeX is running in operational mode to regularly forecast near-term future values of the variable. This forecast constitutes the root mechanism in the early warning mechanism 440.

[0064]   In general, a variable has a time-varying or non-stationary behavior. The models discussed below describe the time-varying behavior at different detail levels. If a time-varying variable is expected to fluctuate around a mean value, then the following simple model may be sufficient,

(1) $S_i = \mu + y_i$,

where Si is the measured value at time index i, μ is the mean of the variable obtained through Least Squares Regression (LSR), assuming uniform time interval, as:

$$(2) \qquad \mu = \frac{\sum_i S_i}{\sum_i 1} ,$$

and residual y_i is a random variable with a mean of zero. This is a mean plus standard deviation model describing the time-varying data fluctuating around a mean value.

[0065] If a time-varying variable has a pattern within a given period (such as a day) but the variation between the periods (as an example, day-to-day variation) is assumed to be random, then the following model may sufficiently describe the pattern,

$$(3) \qquad S_{il} = \mu + \alpha_i + y_{il} ,$$

where i is the index for the time-of-day, l is the index for the l-th day in the data collected, and α denotes the i-th time-of-day deviation from the overall mean u. The factor α_i is obtained from the LSR as

$$(4) \qquad \alpha_i = \frac{\sum_l S_{il}}{\sum_l 1} - \mu ,$$

and the overall mean is calculated through LSR as

$$(5) \qquad \mu = \frac{\sum_{il} S_{il}}{\sum_{il} 1}$$

[0066] With the above definitions, the residual can be computed as: $y_{il} = S_{il} - \mu - \alpha_i$. A residual is the part of the model attributed to random fluctuations or noise in the pattern associated with the same point.

[0067] If a time-varying variable has a pattern not only within a period (e.g., the intra-day patterns) but also between the periods (e.g., day-to-day within a week -- such as data that has a typical variation from Monday to Friday, on top of the variation within a day), then the

following model may sufficiently describe the pattern, assuming the week-to-week variation is random,

$$(6) \qquad S_{ijl} = \mu + \alpha_i + \beta_j + y_{ijl} \, ,$$

where j is the index for the day-of-week, l is the index for the l-th week in the data collected, and β denotes the j-th day-of-week deviation from the overall mean, computed as

$$(7) \qquad \beta_j = \frac{\sum\limits_{il} S_{ijl}}{\sum\limits_{il} 1} - \mu$$

and the time-of-day pattern and overall mean are calculated through LSR as

$$(8) \qquad \alpha_i = \frac{\sum\limits_{jl} S_{ijl}}{\sum\limits_{jl} 1} - \mu$$

$$(9) \qquad \mu = \frac{\sum\limits_{ijl} S_{ijl}}{\sum\limits_{ijl} 1}$$

[0068]   The residual may then be computed as: $y_{ijl} = S_{ijl} - \mu - \alpha_i - \beta_j$.

[0069]   Such modeling may be easily extended to larger time periods. For example, it may be extended to week-of-month effects. In this case, an additional parameter may be used to characterize the k-th week-of-month deviation, denoted by γ_k. This may be necessary for some data that has a structured variation from the first week to the last week of the month, on top the time-of-day and day-of-week variation, assuming the month-to-month variation is random.

[0070]   Different models described above use indices i, j, k, l, that correspond to time. That implies that given any time reference point t, when a variable is measured, the time reference point t may need to be translated into the corresponding indices, i, j, k, l, depending on the specific model used. Using the time reference, the random variables (representing residuals) $y_i, y_{il}, y_{ijl},$ may be uniformly denoted by y_t.

[0071]   As stated earlier, the offline normal behavior modeling mechanism 810 is used to learn the static and regular behavior of a variable. To derive a model that characterize only the regular behavior of a variable based on measured data points (which embed noise), a noise factor may need to be identified and removed from the data points. In addition, an autocorrelation relationship may exist among adjacent data points. That is, y_t may not be an independent and identical distributed (i.i.d) random variable. This property of y_t may further complicate the model. The following model may be applied to remove possible autocorrelation in y_t.

$$(10) \qquad y_t + a_1 y_{t-1} + \ldots + a_p y_{t-p} = \sigma \times u_t,$$

[0072]   The above equation captures the dependency between y_t and the same residuals measured at p previous time reference points. The equation 10 characterizes the p-order autoregressive (AR) process. Let $\hat{\mathbf{a}}^T = \{1, a_1, a_2, \cdots, a_p\}$ be the (p+1)-dimensional AR parameter vector, and u_t is an uncorrelated normal distributed random variable with zero mean and variance of 1 (white noise), and σ is the standard deviation. Let $\hat{\mathbf{a}}^T = \{a_1, a_2, \cdots, a_p\}$ be a p-dimensional vector derived from a^T by deleting its first element; then the covariance estimates of α and the corresponding σ can be calculated by

$$(11) \qquad \hat{\mathbf{a}} = -\mathbf{D}^{-1}\hat{\mathbf{d}}$$

$$(12) \qquad \sigma^2 = \hat{\mathbf{a}}^T \mathbf{C} \hat{\mathbf{a}}$$

[0073]   Here $\mathbf{D}$ is a $p \times p$ submatrix of $\mathbf{C}$ obtained by deleting row and column zeros and $\hat{\mathbf{d}}$ is the p-dimensional vector identical to the first column of $\mathbf{C}$ with the zeroth element deleted. The covariance matrix elements are defined as

$$(13) \qquad c_{ij} = \frac{1}{N'} \sum_{t=p+1}^{N} y_{t-i} y_{t-j}$$

where N is the number of measured points in y_t and $N' = N - p$.

[0074]   According to the description above, the offline normal behavior modeling mechanism 810 establishes an offline normal behavior model for a variable by estimating the model parameters $\mu, \alpha_i, \beta_j, \gamma_k, \{a_1, \cdots, a_p\}, \sigma$ based on given measured data points. During

offline learning, the learning process may be performed in a batch mode using the data points recorded prior to the learning. The learned model, represented by those model parameters, is deployed when the underlying BeX is put in its operational mode.

[0075]   Since the offline normal behavior model does not address (intentionally removes) the dynamics of the variable behavior, the online behavior modeling mechanism 820 may be used to characterize the dynamic behavior of a variable. An online statistical learning mechanism may learn through some window period sliding along the time and may characterize the dynamics using some statistics computed from such sliding windows. The statistics computed from such sliding windows is then compared with the reference window to detect any slow and sudden statistical change in the time series variable. For example, such statistics may include averages or standard deviations.

[0076]   To characterize such dynamic behavior into patterns, it may also be necessary for an online statistical learning mechanism to detect different segments along time in which the statistical properties of the variable dynamics differ significantly. There are known approaches to perform such segmentation based on statistical properties. For example, Generalized Likelihood Ratio (GLR) segmentation does this. When a different segment is identified, the statistics accumulated in the previous segment may need to be replaced with the new statistics accumulated for the new segment. In this way, the online behavior modeling mechanism 820 adaptively, from segment to segment, characterizes the dynamic behavior of a time series variable.

[0077]   Given a normal behavior model for a variable (learned by the offline behavior modeling mechanism 810), the dynamics of the variable behavior can be captured in the residual y_t. In the present invention, the online behavior modeling mechanism 820 utilizes an auto-regression (AR) model to analyze the behavior of y_t defined in equation 10.

[0078]   If a time series residual (variable) is dictated by an auto-correlation statistical property, the AR coefficients $\{a_1, a_2, \cdots, a_p\}$ can be estimated online and dynamically updated over time. Different approaches exist to perform such online estimation and dynamic updating. The identified auto-correlation may be used to predict the future residual values, hence also the variable values. This will facilitate the early warning capability of a BeX in an e-service management system by forecasting that certain threshold violation events may happen, with a certain probability, in the specified time horizon.

[0079]   Prior to updating auto-correlation coefficients, the online behavior modeling mechanism 820 may detect any changes in statistical properties. This is due to the fact that the underlying time series variable (representing the residuals) are often only piecewise stochastically stationary. Therefore, the following two tasks have to be performed during online statistical learning. First, the online behavior modeling mechanism 820 identifies a new segmentation boundary whenever there is a significant statistical property change. Secondly, when the new boundary is identified, the accumulated statistics prior to the new boundary need to be flushed out so that statistical properties for the new segment can be accumulated without the data from a segment that is not statistically coherent. Such segmentation may be implemented using the Generalized Likelihood Ratio method.

[0080]   Fig. 9 is an exemplary flowchart of a process, in which statistical models characterizing the normal and dynamic behavior of a variable are established and are applied in generating early warning of threshold violation in eService management. Offline observation data with respect to a variable is first collected at act 910. The observation data collected offline is assumed to represent the normal behavior of the variable and is used to establish, at act 920, a statistical model that characterizes the normal behavior of the variable. To model the dynamic behavior of the variable, online observation data is collected at act 930 and is used to establish, at act 940, a statistical model that characterizes the dynamic behavior of the variable. The generated models are then used, at act 950, to generate early warning of threshold violation with respect to the variable. Both the established statistical models and the generated early warning are used to detect, at act 960, abnormal behavior of the variable.

[0081]   Fig. 10 is an exemplary flowchart for the online behavior modeling mechanism 820. A new observation is received first at act 1010. The received observation is used to update, at act 1020, a history buffer. The online behavior modeling mechanism 820 then examines, at act 1030, to see whether there are enough observations accumulated to perform learning. If not , the process returns back to act 1010 to collect new observations. If there are enough observations collected for learning, a segmentation is performed, at act 1040, that detects any significant statistical property change that may correspond to a different segment of data.

[0082]   If a new segment is detected, determined at act 1050, the online behavior modeling mechanism 820 identifies, at act 1080, the boundary of the new segment and flushes out, at act 1090, the information that is stored in the history buffer before the detected

- 18 -

new boundary. The process then returns to act 1010 to continue to collect new observations for the new segment. If no new segment is detected, determined at act 1050, the observation data collected so far is used to dynamically estimate (or update), at act 1060, the auto-regression parameters. Such estimated auto-regression parameters are then sent, at act 1070, to the early warning mechanism 440.

[0083]   Based on the regular behavior (learned by the offline normal behavior modeling mechanism 810) and the dynamic behavior (learned dynamically by the online behavior modeling mechanism 820) of a time series variable, the future behavior of the variable may be predicted or forecasted. The certainty with which the future can be predicted may depend on many factors, including the compactness of the underlying patterns (the amount of randomness in the behavior), the depth of the historical base (how much past data is available for pattern discovery), the validity of the modeling techniques adopted, the amount of error in the model (how well the model represents the actual patterns), and how far into the future to predict (the further in the future we predict, the less confidence we have in our prediction).

[0084]   The early warning mechanism 440 (Fig. 4) utilizes the predictive model created during statistical learning (by both offline normal behavior modeling mechanism 810 and the online behavior modeling mechanism 820) to evaluate the direction, magnitude, and rate of change of a BeX variable. In particular, the statistical model for the variable behavior may be used to predict when a critical threshold may be violated. To illustrate, consider a rule used in a BeX:

if X > A then

    SendEvent(S1);

end if

[0085]   The above rule indicates "if the value of X in the current time period exceeds the threshold **A**, then send a violation event". The goal of the early warning mechanism 440 is to predict when (at what of time t in the future) X_t will exceed the threshold **A**. This is illustrated in Fig. 11 and Fig. 12. In Fig. 11(a), the horizontal axis represents the time and the vertical axis represents the magnitude of a variable value. The location of the threshold **A** (1105) is shown in Fig. 11(a) and a curve 1110 represents the actual behavior of variable X as recorded up to the current time 1115 (the dividing point between history and future).

- 19 -

[0086]   In Fig. 11 (b), with the time, the values of variable X are continuously measured and recorded. Such recorded values form a continuing curve 1120. From curve 1120, it can be seen that the values of variable X over time (the behavior of variable X) are steadily trending toward the threshold (note that "steadily trending" is not a requirement of the model, but is used here to simplify the discussion.). In Fig. 11 (b), the movement of X is recorded across the next three analysis intervals (these might correspond to the data sampling rates of the variable) and eventually at the third interval, the variable X exceeds the threshold A and an event may be thrown to indicate that an abnormal event has been detected.

[0087]   The goal of the early warning mechanism 440 is to predict the likelihood of a threshold violation at a specific time in the future and may assign that likelihood a degree of certainty. The statistical model of a variable learned during offline and online statistical learning may be used to facilitate the task. This is illustrated in Fig. 12. When statistical learning is applied to the curve 1110, a statistical model can be derived that characterizes the behavior of variable X based on the data points on curve 1110. Such a statistical model allows the early warning mechanism 440 to look ahead a number of analysis periods and forecast the behavior of the variable X. In Fig. 12, the dotted curve 1250 represents the predicted behavior of variable X in the next three sampling points and a predicted point and time 1240 of threshold violation may also be estimated.

[0088]   In Fig. 13, an exemplary flowchart for the early warning mechanism 440 is described. Given a time reference value t, the early warning mechanism 440 first identifies, at act 1310, the corresponding indices i, j, k (e.g., day, week, month), based on which the residual value at time t or y_t is derived, at act 1320, based on the statistical model of the variable. That is,

$$y_t = \mu - \alpha_i - \beta_j - \gamma_k .$$

[0089]   Using the current value of the residual y_t, the early warning mechanism 440 generates, at act 1330, a forecast of the residual value at a number of future time reference points. For example, to predict the forecast mean of y_t at the future time reference points of $t+1, t+2,..., t+H$, or $t+h$, h-$1,2,...,H$, where H is the maximum prediction horizon, the following computation may be carried out:

$$y_{t+1} = c_1 y_t + c_2 y_{t-1}$$

$$y_{t+2} = c_1 y_{t+1} + c_2 y_t$$

$$\ldots$$

$$y_{t+H} = c_1 y_{t+H-1} + c_2 y_{t+H-2}$$

where second order AR process is assumed here.

[0090]   The early warning mechanism 440 then estimates, at act 1340, the variances of the generated forecasts at time $h = 1, 2, \ldots, H$:

$$\sigma_t^2 = \sigma_u^2 \left[\sum_{n=0}^{h-1} \frac{\left(A_1^{n+1} - A_2^{n+1} \right)^2}{\left(A_1 - A_2 \right)^2} \right]$$

where

$$A_1 A_2 = -c_2, \qquad A_1 + A_2 = c_1.$$

[0091]   In order to predict when, in the future, the variable value will exceeds some variable threshold, the early warning mechanism 440 further estimates the probability that the variable value exceeds the variable threshold at every $h = 1, 2, \ldots, H$. Alternatively, if the threshold for the variable values can be translated into corresponding residual thresholds for the residual values of the variable, the early warning mechanism 440 may also estimate the probability for the residuals to exceed the corresponding residual thresholds derived accordingly for the residuals.

[0092]   Some BeXs may also employ rules enforce that variable values to be within a specific range, defined by two thresholds – a low and a high threshold. In this case, the prediction of a threshold violation may be estimated with respect to both thresholds. Similarly, the prediction of a violation with respect to both low and high variable thresholds may be performed based on residual values using translated low and high thresholds for the residual values.

[0093]   In the exemplary flowchart for the early warning mechanism 440, shown in Fig. 13, a low variable threshold T and a high threshold T' for the variable are translated, at act 1350, into the corresponding low and high residual thresholds (e.g., Th and Th') using the following computation:

$$th'_{t+h} = T' - \mu - \alpha_{i'} - \beta_{j'} - \gamma_{k'}$$
$$th_{t+h} = T - \mu - \alpha_{i'} - \beta_{j'} - \gamma_{k'} \quad , \quad h=1,2,...,H,$$

where (i', j', k') are the indices for $t+h$.

[0094]   Based on derived low and high residual thresholds, the probability for a residual value to remain within the range of $[th, th']$ (or X within $[T,T']$) can be computed, at act 1360, as:

$$P_{t+h} = \Phi\left(\frac{y_{t+h} - th_{t+h}}{\sigma_{t+h}}\right) + \Phi\left(\frac{th'_{t+h} - y_{t+h}}{\sigma_{t+h}}\right), h = 1, 2, ..., H$$

where $\Phi(x)$ is the Cumulative Distribution Function (CDF) of standard normal at x,

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-x^2}$$

[0095]   The probability for the variable to exceed the threshold can be simply derived from $1 - P_{t+h}$.

[0096]   The thresholds (T, T') and the maximum number of future time steps H may be determined by the designer or user of the BeX. The predictive detection system will generate a forecast of the variable values in each future time interval as well as the probability of violating the thresholds. An early warning message may be sent out if the model predicts a threshold violation with a sufficiently high probability (may also be established by the designer or a user).

[0097]   The processing described above may be performed by a general-purpose computer alone or in connection with a special purpose computer. Such processing may be performed by a single platform or by a distributed processing platform. In addition, such processing and functionality can be implemented in the form of special purpose hardware or in the form of software being run by a general-purpose computer. Any data handled in such processing or created as a result of such processing can be stored in any memory as is conventional in the art. By way of example, such data may be stored in a temporary memory, such as in the RAM of a given computer system or subsystem. In addition, or in the alternative, such data may be stored in longer-term storage devices, for example, magnetic disks, rewritable optical disks, and so on. For purposes of the disclosure herein, a computer-

readable media may comprise any form of data storage mechanism, including such existing memory technologies as well as hardware or circuit representations of such structures and of such data.

[0098]   While the invention has been described with reference to the certain illustrated embodiments, the words that have been used herein are words of description, rather than words of limitation. Changes may be made, within the purview of the appended claims, without departing from the scope and spirit of the invention in its aspects. Although the invention has been described herein with reference to particular structures, acts, and materials, the invention is not to be limited to the particulars disclosed, but rather extends to all equivalent structures, acts, and, materials, such as are within the scope of the appended claims.